Fully homomorphic–attribute-based cryptography security using verifiable integrity proofing for secured cloud environment

Ms. P. Tamilselvi[1], Dr.R. Durga[2]

[1]Research Scholar, Computer Science, VISTAS, Chennai, India. E-mail: tamizs2k2@gmail.com [2]Associate Professor, Computer Science, VISTAS, Chennai, India. E-mail: drrdurgaresearch@gmail.com

Abstract

Cloud Computing is the most happening thing in Cloud Environment. Many organizations are accepting and migrating their computational needs to cloud service providers in order to save costs but not at the cost of security. Security is the top most priority for everyone. Many countries follow their own specific guidelines like CCPA and GDPR. In the current Global world, maintain Privacy and meeting the guidelines is a tough job and one such encryption method that satisfies all the requirements is FHE (Fully Homomorphic Encryption). The advantage and disadvantage of FHE is it allows operations on the encrypted data which is not the case with any other encryption methodology. Attribute based Encryption is the most discussed topic and is being widely considered for various data encryption options for better performance and is also one of the most widely researched upon.

Keywords: Cloud Computing, Security, Encryption Standards, Homomorphic Encryption, Access Restriction.

I. Introduction

With the wide operation of pall calculating further and more sensitive information and private data are stored in pall. Furnishing security for data in pall is one of the most important aspects and in order to do that data should be stored in an translated format, which adds an fresh cost. It isn't possible for traditional encryption styles to perform colorful operations on the data by maintaining the data in translated form. Homomorphic encryption addresses these issues and perform colorful operations on translated data by reducing decryption of data at both the ends to perform colorful operations. In this paper, single and completely homomorphic encryption algorithms were described and anatomized independently. A comparison is done with the most generally used Homomorphic encryption algorithms grounded on the performance pointers.

- Incompletely Homomorphic Encryption (PHE) In PHE scheme, only one type of fine operation is allowed on the translated communication, i.e., either addition or addition operation, with unlimited number of times,
- Kindly Homomorphic Encryption (SHE) In SHE, both addition and addition operation is allowed but with only a limited number of times.
- The cryptosystems based on public key encryption attracts a wide range of interest in variety of real time applications due to its robust security. Essentially the public key cryptography is based on factorizing an extended number which has the different nodes with protocol[2] as a

primary operation. These primary operations can be accomplished by in terms of standard floating point operations and a plenty of research works were proposed in the literature including transfer of parameter specification algorithm. Where all the computations were carried out by means of simple shifting and addition operations and require to estimate crypto data permutation tool. It treats to maintain secret and trust communication channel.

Homomorphic Addition
c1 = q1 ∗ p 2 ∗ r1 m1 c2 = q2 ∗ p 2 ∗
r2 m2 c1 c2 = (q1 q2) ∗ p 2 ∗ (r1 r2)
(m1 m2)
Homomorphic Multiplication c1 = q1 ∗ p
2 ∗ r1 m1 c2 = q2 ∗ p 2 ∗ r2 m2 c1 ∗
c2 = ((c1 ∗ q2) q1 ∗ c2 ∗ q1 ∗ q2) ∗
p 2 (2 ∗ r1 ∗ r2 r1 ∗ m2 m1 ∗ r2) m1
∗ m2

RSA ALGORITHM

The public and the private key-generation algorithm is the most complex part of RSA crypto mechanism. Using of prime numbers p and q creates mono Rabin-Miller primary test algorithm. According to the performance of modulo will be calculated by multiplication of p and q. Specification of these numbers are provided by the public keys and making a link between them. Its length usually expressed in bits are called the key length. Both the public and private keys consist of a modulus n operation[5] and the former is associated with an exponent e and the later with an exponent d. The typical value set for these exponents is 65537 and generally is not too large. Further there is no secrecy maintained with the public exponent as it is shared with everyone. However, figure .1 implies the private exponent d is calculated using the Extended algorithm to find the multiplicative inverse with respect to the quotient of n. RSA algorithmic rule is delineate as the system includes a communications channel coupled to a minimum of one terminal[6] having associate secret writing device and to a minimum of one terminal having a decipherment device.

ALGORITHM 1:

RSA (DPSP) Encrypt Decrypt Algorithm:

- RSA secure mechanism secure the data in an open environment;
- Mechanism calculates sizes of the key and messages.

Encryption:

Plain Text Message = (M, Z);

Modulo M = (M, Z);

Decryption:

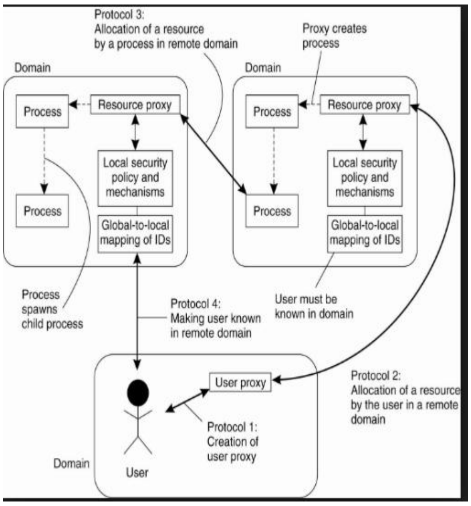
Cipher Text Message = (C, Z);

Modulo N = (C, Z);

M = N modulo M;

N = M modulo N;

Cd = 1+K (N);



while Concrete is confined to lower than 12 bits of perfection (51) (effectively forcing operations to use small plaintext moduli). Also, the rounding crimes that affect from the low- perfection will compound over time for deep operations, as is shown in the delicacy loss reported for deep neural networks in (51). FHEW (39), developed by CWI in Amsterdam, is the precursor of TFHE; still, its bootstrapping speed is inferior to TFHE and there's no support for homomorphic MUX gates. Incipiently, HEAAN (33) implements the CKKS cryptosystem, where both the library and the underpinning scheme are created by the Cryptography Lab at Seoul National University. Still, HEAAN has not seen a major update since 2018 and has been succeeded by other RNS- grounded executions of CKKS in libraries similar as SEAL, PALISADE, and Lattigo.

II.Literature Survey

This section discusses detailed review on the encryption styles used pall data Comparison- Grounded Calculations Over Completely Homomorphic Encrypted Data Mihai Togan1, ∗ and Cezar Ple ̧ scal 1 certSIGN- Exploration and Development, Bucharest, Romania ∗ Corresponding author (E-mailmihai.togan@certsign.ro)

III. Conclusion

In this paper we survey and compare libraries across colorful confines for homomorphic encryption. These ways enable us to perform calculations on translated data as against having to decipher data in order to perform calculations. In this way, it allows for cooperative computing between multiple parties via translated ciphertexts. Although the field is fleetly progressing on the theoretical front, there has been significant recent progress in making it practical from an operation/ practical viewpoint. Both these factors are pivotal for rapid-fire relinquishment and farther development of this field. Operations of homomorphic encryption primarily involve distributed operations in different sectors similar as healthcare, smart grids or genomics. In these operations, ciphertexts, public keys, and other low- position information needs to be participated between data providers, translated computing hosts, and the asked donors of the results of the calculation. 10 A Review of Homomorphic Encryption Libraries for Secure Calculation There are numerous scripts, similar as the one mentioned in healthcare discovery or genomics exploration, where these operations are presently nearly insolvable to develop due to specialized or legal reasons. In cases where the technology is available, one still has to cross the precious and time- consuming hedge of legal processes, driven by the need of maintaining strict sequestration. We can still hope, that practical homomorphic encryption would lead to a dramatic rise in operations in pall and edge calculation where sequestration is critical. Our intent is to partake our literacy, motivate our associates and help the progress of the exploration and technology.

IV. References

1. D. Das, "Secure cloud computing algorithm using homomorphic encryption and multi-party computation," IEEE (ICOIN), 2018, pp. 391-396.
2. Comparison-Based Computations Over Fully Homomorphic Encrypted Data Mihai Togan1,∗ and Cezar Ple¸scal 1 certSIGN -

Research and Development, Bucharest, Romania
∗ Corresponding author (E-mail : mihai.togan@certsign.ro)

3. S. A. Khan, R. K. Aggarwal and S. Kulkarni, "Enhanced Homomorphic Encryption Scheme with PSO for Encryption of Cloud Data," IEEE (ICACCS), 2019, pp. 395-400.
4. 10 Alabdulatif, I. Khalil, A. Y. Zomaya, Z. Tari and X. Yi, "Fully Homomorphic based Privacy-Preserving Distributed Expectation Maximization on Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 31, no. 11, pp. 2668-2681, 1 Nov. 2020, DOI: 10.1109/TPDS.2020.2999407
5. Z. H. Mahmood and M. K. Ibrahem, "New Fully Homomorphic Encryption Scheme Based on Multistage Partial Homomorphic Encryption Applied in Cloud Computing," 2018 1st Annual International Conference on Information and Sciences (AiCIS), Fallujah, Iraq, 2018, pp. 182-186, DOI: 10.1109/AiCIS.2018.
6. K. Rangasami and S. Vagdevi, "Comparative study of homomorphic encryption methods for secured data operations in cloud computing," IEEE (ICEEECCOT), 2017, pp. 1-6. cloud computing," IEEE (ICEEECCOT), 2017, pp. 1-6.
7. R. Kangavalli and Vagdevi S, "A mixed homomorphic encryption scheme for secure data storage incloud," IEEE (IACC), 2015, pp. 1062-1066.
8. D. Das, "Secure cloud computing algorithm using homomorphic encryption and multi-party computation," IEEE (ICOIN), 2018, pp. 391-396.
9. S. A. Khan, R. K. Aggarwal and S. Kulkarni, "Enhanced Homomorphic Encryption Scheme with PSO for Encryption of Cloud Data," IEEE (ICACCS), 2019, pp. 395-400.
10. Encryption Scheme with PSO for Encryption of Cloud Data," IEEE (ICACCS), 2019, pp. 395-400.
11. Alabdulatif, I. Khalil, A. Y. Zomaya, Z. Tari and X. Yi, "Fully Homomorphic based Privacy-Preserving Distributed Expectation Maximization on Cloud," in IEEE Transactions on Parallel and Distributed Systems, vol. 31, no.

11, pp. 2668-2681, 1 Nov. 2020, doi: 10.1109/TPDS.2020.2999407.

12. A. Chatterjee and I. Sengupta, "Translating Algorithms to Handle Fully Homomorphic Encrypted Data on the Cloud," in IEEE Transactions on Cloud Computing, vol. 6, no. 1, pp. 287-300, 1 Jan.-March 2018, doi: 10.1109/TCC.2015.2481416.
13. P. Mishra, V. Varadharajan, E. S. Pilli and U. Tupakula, "VMGuard: A VMI-Based Security Architecture for Intrusion Detection in Cloud Environment," in IEEE Transactions on Cloud Computing, vol. 8, no. 3, pp. 957-971, 1 July-Sept. 2020, doi: 10.1109/TCC.2018.2829202.
14. J. Park, D. S. Kim and H. Lim, "Privacy-Preserving Reinforcement Learning Using Homomorphic Encryption in Cloud Computing Infrastructures," in IEEE Access, vol. 8, pp. 203564-203579, 2020, doi: 10.1109/ACCESS.2020.3036899.
15. O. Alkadi, N. Moustafa and B. Turnbull, "A Review of Intrusion Detection and Blockchain Applications in the Cloud: Approaches, Challenges and Solutions," in IEEE Access, vol. 8, pp. 104893-104917, 2020, doi: 10.1109/ACCESS.2020.2999715.
16. Shashank Bajpai and Padmija Srivastava," A Fully Homomorphic Encryption Implementation on Cloud Computing," in International Journal of Information & Computation Technology (2014).
17. S. Wang, K. Liang, J. K. Liu, J. Chen, J. Yu and W. Xie, "Attribute-Based Data Sharing Scheme Revisited in Cloud Computing," in IEEE Transactions on Information Forensics and Security, vol. 11, no. 8, pp. 1661-1673, Aug. 2016, doi: 10.1109/TIFS.2016.2549004.
18. 15. K. Fan, N. Huang, Y. Wang, H. Li and Y. Yang, "Secure and Efficient Personal Health Record Scheme Using Attribute-Based Encryption," 2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing, New York, NY, USA, 2015, pp. 111-114, doi: 10.1109/CSCloud.2015.40.
19. Kumar, Saravana & G.V, Raya & Balamurugan, Balamurugan. (2014). Enhanced Attribute Based Encryption for Cloud Computing. Procedia Computer Science. 46. 10.1016/j.procs.2015.02.127.

20. 2. Kumar, P. Praveen; Kumar, P. Syam; Alphonse, P.J.A. (2018). Attribute based encryption in cloud computing: A survey, gap analysis, and future directions. Journal of Network and Computer Applications, (), S1084804518300547–. doi:10.1016/j.jnca.2018.02.009